



Terms used

[fragment](#) [delay](#) [conflict](#) [position](#) [tile](#) [token](#) [shading](#) [shader](#)

Found 16 of 171,143

Sort results by

relevance

Display results

expanded form


[Save results to a Binder](#)

[Search Tips](#)
☐ Open results in a new window

Try an [Advanced Search](#)

Try this search in [The ACM Guide](#)

Results 1 - 16 of 16

Relevance scale 


### 1 [Real-time shading](#)



Marc Olano, Kurt Akeley, John C. Hart, Wolfgang Heidrich, Michael McCool, Jason L. Mitchell, Randi Rost

August 2004 **Proceedings of the conference on SIGGRAPH 2004 course notes GRAPH '04**

Publisher: ACM Press

Full text available:  pdf(7.39 MB)

Additional Information: [full citation](#), [abstract](#)

Real-time procedural shading was once seen as a distant dream. When the first version of this course was offered four years ago, real-time shading was possible, but only with one-of-a-kind hardware or by combining the effects of tens to hundreds of rendering passes. Today, almost every new computer comes with graphics hardware capable of interactively executing shaders of thousands to tens of thousands of instructions. This course has been redesigned to address today's real-time shading capabilities ...


### 2 [Realizing OpenGL: two implementations of one architecture](#)



Mark J. Kilgard

August 1997 **Proceedings of the ACM SIGGRAPH/EUROGRAPHICS workshop on Graphics hardware**

Publisher: ACM Press

Full text available:  pdf(1.66 MB)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)
**Keywords:** O2, OpenGL, graphics hardware architecture, infinite-reality


### 3 [Special issue on knowledge representation](#)



Ronald J. Brachman, Brian C. Smith

February 1980 **ACM SIGART Bulletin**, Issue 70

Publisher: ACM Press

Full text available:  pdf(13.13 MB)

Additional Information: [full citation](#), [abstract](#)

In the fall of 1978 we decided to produce a special issue of the SIGART Newsletter devoted to a survey of current knowledge representation research. We felt that there were two useful functions such an issue could serve. First, we hoped to elicit a clear picture of how people working in this subdiscipline understand knowledge representation research, to illuminate the issues on which current research is focused, and to catalogue what approaches and techniques are currently being developed. Second ...

### 4 [Seeing, hearing, and touching: putting it all together](#)



Brian Fisher, Sidney Fels, Karon MacLean, Tamara Munzner, Ronald Rensink

August 2004 **Proceedings of the conference on SIGGRAPH 2004 course notes GRAPH '04**


5 [A structural view of the Cedar programming environment](#)



Daniel C. Swinehart, Polle T. Zellweger, Richard J. Beach, Robert B. Hagmann

August 1986 **ACM Transactions on Programming Languages and Systems (TOPLAS)**,  
Volume 8 Issue 4

Publisher: ACM Press

Full text available:  pdf(6.32 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citings](#), [index terms](#)

This paper presents an overview of the Cedar programming environment, focusing on its overall structure—that is, the major components of Cedar and the way they are organized. Cedar supports the development of programs written in a single programming language, also called Cedar. Its primary purpose is to increase the productivity of programmers whose activities include experimental programming and the development of prototype software systems for a high-performance personal computer. T ...

6 [Exploiting weak connectivity for mobile file access](#)



L. B. Mummert, M. R. Ebling, M. Satyanarayanan

December 1995 **ACM SIGOPS Operating Systems Review , Proceedings of the fifteenth ACM symposium on Operating systems principles SOSP '95**, Volume 29  
Issue 5

Publisher: ACM Press

Full text available:  pdf(1.49 MB)

Additional Information: [full citation](#), [references](#), [citings](#), [index terms](#)

7 [Dynamically scheduled VLIW processors](#)

B. Ramakrishna Rau

December 1993 **Proceedings of the 26th annual international symposium on Microarchitecture**

Publisher: IEEE Computer Society Press

Full text available:  pdf(1.64 MB)

Additional Information: [full citation](#), [references](#), [citings](#)

**Keywords:** VLIW processors, dynamic scheduling, multiple operation issue, out-of-order execution, scoreboarding

8 [Fortran 8X draft](#)



Loren P. Meissner

December 1989 **ACM SIGPLAN Fortran Forum**, Volume 8 Issue 4

Publisher: ACM Press

Full text available:  pdf(21.36 MB)

Additional Information: [full citation](#), [abstract](#), [index terms](#)

**Standard Programming Language Fortran.** This standard specifies the form and establishes the interpretation of programs expressed in the Fortran language. It consists of the specification of the language Fortran. No subsets are specified in this standard. The previous standard, commonly known as "FORTRAN 77", is entirely contained within this standard, known as "Fortran 8x". Therefore, any standard-conforming FORTRAN 77 program is standard conforming under this standard. New features can b ...

9 [Distributed file systems: concepts and examples](#)



Eliezer Levy, Abraham Silberschatz

December 1990 **ACM Computing Surveys (CSUR)**, Volume 22 Issue 4

Publisher: ACM Press

Full text available:  pdf(5.33 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citings](#), [index terms](#), [review](#)

The purpose of a distributed file system (DFS) is to allow users of physically distributed computers to share data and storage resources by using a common file system. A typical

configuration for a DFS is a collection of workstations and mainframes connected by a local area network (LAN). A DFS is implemented as part of the operating system of each of the connected computers. This paper establishes a viewpoint that emphasizes the dispersed structure and decentralization of both data and con ...

## 10 Distributed operating systems



Andrew S. Tanenbaum, Robbert Van Renesse

December 1985 **ACM Computing Surveys (CSUR)**, Volume 17 Issue 4

**Publisher:** ACM Press

Full text available: [pdf\(5.49 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Distributed operating systems have many aspects in common with centralized ones, but they also differ in certain ways. This paper is intended as an introduction to distributed operating systems, and especially to current university research about them. After a discussion of what constitutes a distributed operating system and how it is distinguished from a computer network, various key design issues are discussed. Then several examples of current research projects are examined in some detail ...

## 11 Fault-tolerance in the advanced automation system



Flaviu Cristian, Bob Dancey, Jon Dehn

September 1990 **Proceedings of the 4th workshop on ACM SIGOPS European workshop**

**Publisher:** ACM Press

Full text available: [pdf\(1.39 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)

The Advanced Automation System is a distributed real-time system under development by IBM's Systems Integration Division for the US Federal Aviation Administration. The system is intended to replace the present en-route and terminal approach US air traffic control computer systems over the next decade. High availability of air traffic control services is an essential requirement of the system. This paper discusses the general approach to fault-tolerance adopted in AAS, by reviewing some of the q ...

## 12 Automating the lexical and syntactic design of graphical user interfaces: the UofA\*



**UIMS**

Gurminder Singh, Mark Green

July 1991 **ACM Transactions on Graphics (TOG)**, Volume 10 Issue 3

**Publisher:** ACM Press

Full text available: [pdf\(3.82 MB\)](#)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#), [review](#)

## 13 PAMELA: a rule-based AI language for process-control applications



F. Barachini

June 1988 **Proceedings of the 1st international conference on Industrial and engineering applications of artificial intelligence and expert systems - Volume 2 IEA/AIE '88**

**Publisher:** ACM Press

Full text available: [pdf\(652.36 KB\)](#)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

## 14 Special section: Reasoning about structure, behavior and function



B. Chandrasekaran, Rob Milne

July 1985 **ACM SIGART Bulletin**, Issue 93

**Publisher:** ACM Press

Full text available: [pdf\(5.13 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#)

The last several years' of work in the area of knowledge-based systems has resulted in a deeper understanding of the potentials of the current generation of ideas, but more importantly, also about their limitations and the need for research both in a broader framework as well as in new directions. The following ideas seem to us to be worthy of note in this connection.

15 Draft report on requirements for a common prototyping system



R. P. Gabriel

March 1989 **ACM SIGPLAN Notices**, Volume 24 Issue 3

**Publisher:** ACM Press

Full text available: pdf(4.76 MB)

Additional Information: [full citation](#), [citations](#), [index terms](#)



16 Modeling for text compression



Timothy Bell, Ian H. Witten, John G. Cleary

December 1989 **ACM Computing Surveys (CSUR)**, Volume 21 Issue 4

**Publisher:** ACM Press

Full text available: pdf(3.54 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)



The best schemes for text compression use large models to help them predict which characters will come next. The actual next characters are coded with respect to the prediction, resulting in compression of information. Models are best formed adaptively, based on the text seen so far. This paper surveys successful strategies for adaptive modeling that are suitable for use in practical text compression systems. The strategies fall into three main classes: finite-context modeling, i ...

Results 1 - 16 of 16

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2006 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads: [Adobe Acrobat](#) [QuickTime](#) [Windows Media Player](#) [Real Player](#)



Terms used

[fragment](#) [delay](#) [conflict](#) [position](#) [tile](#) [shading](#) [shader](#)

Found 62 of 171,143

Sort results by


[Save results to a Binder](#)

Display results


[Search Tips](#)
☐ Open results in a new window

Try an [Advanced Search](#)

Try this search in [The ACM Guide](#)

Results 1 - 20 of 62

Result page: [1](#) [2](#) [3](#) [4](#) [next](#)

Relevance scale ☐ ☐ ☐ ☐ ☐

### 1 [Delay streams for graphics hardware](#)



Timo Aila, Ville Miettinen, Petri Nordlund

July 2003 **ACM Transactions on Graphics (TOG)**, Volume 22 Issue 3

Publisher: ACM Press

Full text available:  pdf(1.67 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

In causal processes decisions do not depend on future data. Many well-known problems, such as occlusion culling, order-independent transparency and edge antialiasing cannot be properly solved using the traditional causal rendering architectures, because future data may change the interpretation of current events. We propose adding a *delay stream* between the vertex and pixel processing units. While a triangle resides in the delay stream, subsequent triangles generate occlusion information. ...

**Keywords:** 3D graphics hardware, antialiasing, occlusion culling, order-independent transparency, stream processing


### 2 [Real-time shading](#)



Marc Olano, Kurt Akeley, John C. Hart, Wolfgang Heidrich, Michael McCool, Jason L. Mitchell, Randi Rost

August 2004 **Proceedings of the conference on SIGGRAPH 2004 course notes GRAPH '04**

Publisher: ACM Press

Full text available:  pdf(7.39 MB)

Additional Information: [full citation](#), [abstract](#)

Real-time procedural shading was once seen as a distant dream. When the first version of this course was offered four years ago, real-time shading was possible, but only with one-of-a-kind hardware or by combining the effects of tens to hundreds of rendering passes. Today, almost every new computer comes with graphics hardware capable of interactively executing shaders of thousands to tens of thousands of instructions. This course has been redesigned to address today's real-time shading capabili ...

### 3 [Realizing OpenGL: two implementations of one architecture](#)



Mark J. Kilgard

August 1997 **Proceedings of the ACM SIGGRAPH/EUROGRAPHICS workshop on Graphics hardware**

Publisher: ACM Press

Full text available:  pdf(1.66 MB)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

**Keywords:** O2, OpenGL, graphics hardware architecture, infinite-reality



## Special issue on knowledge representation

Ronald J. Brachman, Brian C. Smith  
February 1980 **ACM SIGART Bulletin**, Issue 70

**Publisher:** ACM Press

Full text available: [pdf\(13.13 MB\)](#) Additional Information: [full citation](#), [abstract](#)

In the fall of 1978 we decided to produce a special issue of the SIGART Newsletter devoted to a survey of current knowledge representation research. We felt that there were two useful functions such an issue could serve. First, we hoped to elicit a clear picture of how people working in this subdiscipline understand knowledge representation research, to illuminate the issues on which current research is focused, and to catalogue what approaches and techniques are currently being developed. Second ...

## 5 A Characterization of Ten Hidden-Surface Algorithms



Evan E. Sutherland, Robert F. Sproull, Robert A. Schumacker  
March 1974 **ACM Computing Surveys (CSUR)**, Volume 6 Issue 1

**Publisher:** ACM Press

Full text available: [pdf\(4.47 MB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

## 6 Seeing, hearing, and touching: putting it all together



Brian Fisher, Sidney Fels, Karon MacLean, Tamara Munzner, Ronald Rensink  
August 2004 **Proceedings of the conference on SIGGRAPH 2004 course notes GRAPH '04**

**Publisher:** ACM Press

Full text available: [pdf\(20.64 MB\)](#) Additional Information: [full citation](#)

## 7 Data space-oriented tiling for enhancing locality



I. Kadayif, M. Kandemir  
May 2005 **ACM Transactions on Embedded Computing Systems (TECS)**, Volume 4 Issue 2

**Publisher:** ACM Press

Full text available: [pdf\(978.44 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Improving locality of data references is becoming increasingly important due to increasing gap between processor cycle times and off-chip memory access latencies. Improving data locality not only improves effective memory access time but also reduces memory system energy consumption due to data references. An optimizing compiler can play an important role in enhancing data locality in array-intensive embedded media applications with regular data access patterns. This paper presents a compiler-based ...

**Keywords:** Software compilation, array-intensive applications, data locality, iteration space tiling, scratch pad memory

## 8 A structural view of the Cedar programming environment



Daniel C. Swinehart, Polle T. Zellweger, Richard J. Beach, Robert B. Hagmann  
August 1986 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 8 Issue 4

**Publisher:** ACM Press

Full text available: [pdf\(6.32 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

This paper presents an overview of the Cedar programming environment, focusing on its overall structure—that is, the major components of Cedar and the way they are organized. Cedar supports the development of programs written in a single programming language, also called Cedar. Its primary purpose is to increase the productivity of programmers whose activities include experimental programming and the development of prototype software systems for a high-performance personal computer. This ...

## 9 Virtual Memory

Peter J. Denning



**Publisher:** ACM Press

Full text available: [pdf\(2.63 MB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

## 10 High-level synthesis of distributed logic-memory architectures



Chao Huang, Srivaths Ravi, Anand Raghunathan, Niraj K. Jha

November 2002 **Proceedings of the 2002 IEEE/ACM international conference on Computer-aided design**

**Publisher:** ACM Press

Full text available: [pdf\(1.10 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

With the increasing cost of global communication on-chip, high-performance designs for data-intensive applications require architectures that distribute hardware resources (computing logic, memories, interconnect, etc.) throughout a chip, while restricting computations and communications to geographic proximities. In this paper, we present a methodology for high-level synthesis (HLS) of distributed logic-memory architectures, i.e., architectures that have logic and memory distribut ...

## 11 Exploiting weak connectivity for mobile file access



L. B. Mummert, M. R. Ebling, M. Satyanarayanan

December 1995 **ACM SIGOPS Operating Systems Review , Proceedings of the fifteenth ACM symposium on Operating systems principles SOSP '95**, Volume 29 Issue 5

**Publisher:** ACM Press

Full text available: [pdf\(1.49 MB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

## 12 Query evaluation techniques for large databases



Goetz Graefe

June 1993 **ACM Computing Surveys (CSUR)**, Volume 25 Issue 2

**Publisher:** ACM Press

Full text available: [pdf\(9.37 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Database management systems will continue to manage large data volumes. Thus, efficient algorithms for accessing and manipulating large sets and sequences will be required to provide acceptable performance. The advent of object-oriented and extensible database systems will not solve this problem. On the contrary, modern data models exacerbate the problem: In order to manipulate large sets of complex objects as efficiently as today's database systems manipulate simple records, query-processi ...

**Keywords:** complex query evaluation plans, dynamic query evaluation plans, extensible database systems, iterators, object-oriented database systems, operator model of parallelization, parallel algorithms, relational database systems, set-matching algorithms, sort-hash duality

## 13 Hybrid volume and polygon rendering with cube hardware



Kevin Kreeger, Arie Kaufman

July 1999 **Proceedings of the ACM SIGGRAPH/EUROGRAPHICS workshop on Graphics hardware**

**Publisher:** ACM Press

Full text available: [pdf\(1.85 MB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

**Keywords:** cube architecture, mixing polygons and volumes, ray casting, run-length-encoding, volume rendering

## 14

RAID: high-performance, reliable secondary storage





**Publisher:** ACM Press

Full text available: pdf(3.60 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Disk arrays were proposed in the 1980s as a way to use parallelism between multiple disks to improve aggregate I/O performance. Today they appear in the product lines of most major computer manufacturers. This article gives a comprehensive overview of disk arrays and provides a framework in which to organize current and future work. First, the article introduces disk technology and reviews the driving forces that have popularized disk arrays: performance and reliability. It discusses the tw ...

**Keywords:** RAID, disk array, parallel I/O, redundancy, storage, striping

15 Dynamically scheduled VLIW processors

B. Ramakrishna Rau

December 1993 **Proceedings of the 26th annual international symposium on Microarchitecture**

**Publisher:** IEEE Computer Society Press

Full text available: pdf(1.64 MB)

Additional Information: [full citation](#), [references](#), [citations](#)

**Keywords:** VLIW processors, dynamic scheduling, multiple operation issue, out-of-order execution, scoreboarding

16 Code optimization II: Code optimization for code compression

Milenko Drinić, Darko Kirovski, Hoi Vo

March 2003 **Proceedings of the international symposium on Code generation and optimization: feedback-directed and runtime optimization CGO '03**

**Publisher:** IEEE Computer Society

Full text available: pdf(1.07 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

With the emergence of software delivery platforms such as Microsoft's .NET, reduced size of transmitted binaries has become a very important system parameter strongly affecting system performance. In this paper, we present two novel pre-processing steps for code compression that explore program binaries' syntax and semantics to achieve superior compression ratios. The first preprocessing step involves heuristic partitioning of a program binary into streams with high auto-correlation. The second ...

17 Zero-cycle loads: microarchitecture support for reducing load latency

Todd M. Austin, Gurindar S. Sohi

December 1995 **Proceedings of the 28th annual international symposium on Microarchitecture**

**Publisher:** IEEE Computer Society Press

Full text available: pdf(1.35 MB)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

18 TraceBack: first fault diagnosis by reconstruction of distributed control flow



Andrew Ayers, Richard Schooler, Chris Metcalf, Anant Agarwal, Junghwan Rhee, Emmett Witchel

June 2005 **ACM SIGPLAN Notices , Proceedings of the 2005 ACM SIGPLAN conference on Programming language design and implementation PLDI '05**, Volume 40 Issue 6

**Publisher:** ACM Press

Full text available: pdf(347.77 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Faults that occur in production systems are the most important faults to fix, but most production systems lack the debugging facilities present in development environments. TraceBack provides debugging information for production systems by providing execution history data about program problems (such as crashes, hangs, and exceptions).



**Keywords:** fault diagnosis, instrumentation

## 19 Automatic data and computation decomposition on distributed memory parallel



### computers

Peizong Lee, Zvi Meir Kedem

January 2002 **ACM Transactions on Programming Languages and Systems (TOPLAS)**,  
Volume 24 Issue 1

**Publisher:** ACM Press

Full text available: pdf(1.15 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

To exploit parallelism on shared memory parallel computers (SMPCs), it is natural to focus on decomposing the computation (mainly by distributing the iterations of the nested Do-Loops). In contrast, on distributed memory parallel computers (DMPCs), the decomposition of computation and the distribution of data must both be handled---in order to balance the computation load and to minimize the migration of data. We propose and validate experimentally a method for handling computations and data syn ...

**Keywords:** Computation decomposition, data alignment, data distribution, distributed-memory computers, dominant data array, iteration space mapping vector, parallelizing compilers, spatial dependence vector, temporal dependence vector, tiling techniques

## 20 The Zebra striped network file system



John H. Hartman, John K. Ousterhout

December 1993 **ACM SIGOPS Operating Systems Review , Proceedings of the fourteenth ACM symposium on Operating systems principles SOSP '93**, Volume 27 Issue 5

**Publisher:** ACM Press

Full text available: pdf(1.93 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Zebra is a network file system that increases throughput by striping file data across multiple servers. Rather than striping each file separately, Zebra forms all the new data from each client into a single stream, which it then stripes using an approach similar to a log-structured file system. This provides high performance for writes of small files as well as for reads and writes of large files. Zebra also writes parity information in each stripe in the style of RAID disk arrays; this increase ...

Results 1 - 20 of 62

Result page: [1](#) [2](#) [3](#) [4](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2006 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads: [Adobe Acrobat](#) [QuickTime](#) [Windows Media Player](#) [Real Player](#)



Terms used

[fragment](#) [stall](#) [conflict](#) [position](#) [tile](#) [token](#) [shading](#) [shader](#)

Found 6 of 171,143

Sort results by

relevance

Display results

expanded form

 [Save results to a Binder](#)
 [Search Tips](#)
☐ [Open results in a new window](#)

Try an [Advanced Search](#)

Try this search in [The ACM Guide](#)

Results 1 - 6 of 6

Relevance scale 


## 1 [Real-time shading](#)



Marc Olano, Kurt Akeley, John C. Hart, Wolfgang Heidrich, Michael McCool, Jason L. Mitchell, Randi Rost

August 2004 **Proceedings of the conference on SIGGRAPH 2004 course notes GRAPH '04**

Publisher: ACM Press

Full text available:  [pdf\(7.39 MB\)](#)

Additional Information: [full citation](#), [abstract](#)

Real-time procedural shading was once seen as a distant dream. When the first version of this course was offered four years ago, real-time shading was possible, but only with one-of-a-kind hardware or by combining the effects of tens to hundreds of rendering passes. Today, almost every new computer comes with graphics hardware capable of interactively executing shaders of thousands to tens of thousands of instructions. This course has been redesigned to address today's real-time shading capabili ...


## 2 [Realizing OpenGL: two implementations of one architecture](#)



Mark J. Kilgard

August 1997 **Proceedings of the ACM SIGGRAPH/EUROGRAPHICS workshop on Graphics hardware**

Publisher: ACM Press


Full text available:  [pdf\(1.66 MB\)](#)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)
**Keywords:** O2, OpenGL, graphics hardware architecture, infinite-reality

## 3 [Proceedings of the SIGNUM conference on the programming environment for development of numerical software](#)


March 1979 **ACM SIGNUM Newsletter**, Volume 14 Issue 1

Publisher: ACM Press

Full text available:  [pdf\(5.02 MB\)](#)


Additional Information: [full citation](#)

## 4 [Dynamically scheduled VLIW processors](#)

B. Ramakrishna Rau

December 1993 **Proceedings of the 26th annual international symposium on Microarchitecture**

Publisher: IEEE Computer Society Press

Full text available:  [pdf\(1.64 MB\)](#)

Additional Information: [full citation](#), [references](#), [citations](#)
**Keywords:** VLIW processors, dynamic scheduling, multiple operation issue, out-of-order

5 Automating the lexical and syntactic design of graphical user interfaces: the UofA\*



**UIMS**

Gurminder Singh, Mark Green

July 1991 **ACM Transactions on Graphics (TOG)**, Volume 10 Issue 3

**Publisher:** ACM Press

Full text available: [pdf\(3.82 MB\)](#)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#),  
[review](#)



6 Topical papers with demonstrations: On text coherence parsing

Udo Hahn

August 1992 **Proceedings of the 14th conference on Computational linguistics - Volume 1**

**Publisher:** Association for Computational Linguistics

Full text available: [pdf\(707.80 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#)

In this paper global patterns of thematic text organization are considered within the framework of a distributed model of text understanding. Based on the parsing results of prior text cohesion analysis, specialized text grammar modules determine whether some well-defined text macro-organization pattern is computable from the available text representation structures. The model underlying text coherence parsing formalizes hitherto entirely intuitive textlinguistic notions whose origin can be tra ...



Results 1 - 6 of 6

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2006 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads: [Adobe Acrobat](#) [QuickTime](#) [Windows Media Player](#) [Real Player](#)



USPTO

[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

Search: ☒ The ACM Digital Library ☐ The Guide



THE ACM DIGITAL LIBRARY



[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

Terms used

**fragment postpone conflict position tile token shading shader**

Found 3 of 171,143

Sort results  
by

relevance



[Save results to a Binder](#)

Display  
results

expanded form



[Search Tips](#)

☐ Open results in a new window

Try an [Advanced Search](#)

Try this search in [The ACM Guide](#)

Results 1 - 3 of 3

Relevance scale ☐ ☐ ☐ ☐ ☐

- 1 [Proceedings of the SIGNUM conference on the programming environment for development of numerical software](#)



March 1979 **ACM SIGNUM Newsletter**, Volume 14 Issue 1

**Publisher:** ACM Press



Terms used

[fragment](#) [postpone](#) [conflict](#) [position](#) [tile](#) [shading](#) [shader](#)

Found 12 of 171,143

Sort results by

relevance

Display results

expanded form

 [Save results to a Binder](#)
 [Search Tips](#)
☐ Open results in a new window

Try an [Advanced Search](#)

Try this search in [The ACM Guide](#)

Results 1 - 12 of 12

Relevance scale ☐ ☐ ☐ ☐ ☐


## 1 [A Characterization of Ten Hidden-Surface Algorithms](#)



Evan E. Sutherland, Robert F. Sproull, Robert A. Schumacker

March 1974 **ACM Computing Surveys (CSUR)**, Volume 6 Issue 1

Publisher: ACM Press


Full text available:  pdf(4.47 MB)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)


## 2 [Proceedings of the SIGNUM conference on the programming environment for development of numerical software](#)


March 1979 **ACM SIGNUM Newsletter**, Volume 14 Issue 1

Publisher: ACM Press

Full text available:  pdf(5.02 MB)

Additional Information: [full citation](#)



## 3 [Special section: Reasoning about structure, behavior and function](#)



B. Chandrasekaran, Rob Milne

July 1985 **ACM SIGART Bulletin**, Issue 93

Publisher: ACM Press

Full text available:  pdf(5.13 MB)

Additional Information: [full citation](#), [abstract](#), [references](#)


The last several years' of work in the area of knowledge-based systems has resulted in a deeper understanding of the potentials of the current generation of ideas, but more importantly, also about their limitations and the need for research both in a broader framework as well as in new directions. The following ideas seem to us to be worthy of note in this connection.


## 4 [Draft report on requirements for a common prototyping system](#)



R. P. Gabriel

March 1989 **ACM SIGPLAN Notices**, Volume 24 Issue 3

Publisher: ACM Press

Full text available:  pdf(4.76 MB)

Additional Information: [full citation](#), [citations](#), [index terms](#)



## 5 [Toward relaxing assumptions in languages and their implementations](#)



Mary Shaw, Wm. A. Wulf

March 1980 **ACM SIGPLAN Notices**, Volume 15 Issue 3

Publisher: ACM Press

Full text available:  pdf(1.47 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)


Language implementors frequently make pre-emptive decisions concerning the exact

implementations of language features. These decisions constrain programmers' control over their computations and may tempt them to write involuted code to obtain special (or efficient) effects. In many cases, we can distinguish some properties of a language facility that are essential to the semantics and other properties that are incidental. Recent abstraction techniques emphasize dealing with such distinctions by ...

## 6 Reducing data cache leakage energy using a compiler-based approach



Wei Zhang, Mahmut Kandemir, Mustafa Karakoy, Guangyu Chen

August 2005 **ACM Transactions on Embedded Computing Systems (TECS)**, Volume 4 Issue 3

**Publisher:** ACM Press

Full text available: [pdf\(750.57 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Silicon technology advances have made it possible to pack millions of transistors---switching at high clock speeds---on a single chip. While these advances bring unprecedented performance to electronic products, they also pose difficult power/energy consumption problems. For example, large number of transistors in dense on-chip cache memories consume significant static (leakage) power even if the cache is not used by the current computation. While previous compiler research studied code and data ...

**Keywords:** Compiler analysis, array-intensive applications, data caches, energy optimization, pointer-intensive applications

## 7 Type theories and object-oriented programming



Scott Danforth, Chris Tomlinson

March 1988 **ACM Computing Surveys (CSUR)**, Volume 20 Issue 1

**Publisher:** ACM Press

Full text available: [pdf\(4.39 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Object-oriented programming is becoming a popular approach to the construction of complex software systems. Benefits of object orientation include support for modular design, code sharing, and extensibility. In order to make the most of these advantages, a type theory for objects and their interactions should be developed to aid checking and controlled derivation of programs and to support early binding of code bodies for efficiency. As a step in this direction, this paper surveys a number ...

## 8 Designing complex systems—a structured activity



Gerrit C. van der Veer, Johannes C. van Vliet, Bert F. Lenting

August 1995 **Proceedings of the conference on Designing interactive systems: processes, practices, methods, & techniques**

**Publisher:** ACM Press

Full text available: [pdf\(1.02 MB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

## 9 Dynamic compilation for energy adaptation



P. Unnikrishnan, G. Chen, M. Kandemir, D. R. Mudgett

November 2002 **Proceedings of the 2002 IEEE/ACM international conference on Computer-aided design**

**Publisher:** ACM Press

Full text available: [pdf\(156.76 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

While previous compiler research indicates that significant improvements in energy efficiency may be possible if properly optimized code is used, the energy constraints under which a given application code should be optimized may not always be available at compile-time. More importantly, these constraints may change dynamically during the course of execution. In this work, we present a dynamic recompilation/linking framework using which the energy behavior of a given application can be optimized ...

## 10 Limitations of synchronous communication with static process structure in languages for distributed computing



Barbara Liskov, Maurice Herlihy, Lucy Gilbert

**Publisher:** ACM Press

Full text available:  [pdf\(1.48 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)

Modules in a distributed program are active, communicating entities. A language for distributed programs must choose a set of communication primitives and a structure for processes. This paper examines one possible choice: synchronous communication primitives (such as rendez-vous or remote procedure call) in combination with modules that encompass a fixed number of processes (such as Ada tasks or UNIX processes). An analysis of the concurrency requirements of distributed programs suggests that t ...


**11** Register allocation with instruction scheduling



Shlomit S. Pinter

June 1993 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1993 conference  
on Programming language design and implementation PLDI '93**, Volume 28  
Issue 6

**Publisher:** ACM Press

Full text available:  [pdf\(931.91 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

We present a new framework in which considerations of both register allocation and instruction scheduling can be applied uniformly and simultaneously. In this framework an optimal coloring of a graph, called the parallel interference graph, provides an optimal register allocation and preserves the property that no false dependences are introduced, thus all the options for parallelism are kept for the scheduler to handle. For this framework we provide heuristics for trading ...


**12** Dynamic memory disambiguation for array references



David Bernstein, Doron Cohen, Dror E. Maydan

November 1994 **Proceedings of the 27th annual international symposium on  
Microarchitecture**

**Publisher:** ACM Press





Full text available:  [pdf\(924.07 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Memory disambiguation, or alias analysis, is a key component of modern optimizing compilers. Any optimization that reorders or changes code containing memory operations must analyze the memory references to ensure that the original semantics of the program are not changed. The recent proliferation of machines able to exploit parallelism, either at the coarse grain or more commonly at the instruction level, has led to the development of sophisticated memory disambiguation algorithm ...

Results 1 - 12 of 12

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2006 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)


Terms used [fragment](#) [stall](#) [conflict](#) [position](#) [tile](#) [shading](#) [shader](#)

Found 30 of 171,143

Sort results by

Display results

☒ [Save results to a Binder](#)
☒ [Search Tips](#)
☐ [Open results in a new window](#)

Try an [Advanced Search](#)

Try this search in [The ACM Guide](#)

Results 1 - 20 of 30

Result page: [1](#) [2](#) [next](#)

Relevance scale ☐ ☐ ☐ ☐ ☐


### 1 [Real-time shading](#)



Marc Olano, Kurt Akeley, John C. Hart, Wolfgang Heidrich, Michael McCool, Jason L. Mitchell, Randi Rost

August 2004 **Proceedings of the conference on SIGGRAPH 2004 course notes GRAPH '04**

Publisher: ACM Press

Full text available:  [pdf\(7.39 MB\)](#)

Additional Information: [full citation](#), [abstract](#)

Real-time procedural shading was once seen as a distant dream. When the first version of this course was offered four years ago, real-time shading was possible, but only with one-of-a-kind hardware or by combining the effects of tens to hundreds of rendering passes. Today, almost every new computer comes with graphics hardware capable of interactively executing shaders of thousands to tens of thousands of instructions. This course has been redesigned to address today's real-time shading capabili ...


### 2 [Realizing OpenGL: two implementations of one architecture](#)



Mark J. Kilgard

August 1997 **Proceedings of the ACM SIGGRAPH/EUROGRAPHICS workshop on Graphics hardware**

Publisher: ACM Press

Full text available:  [pdf\(1.66 MB\)](#)

Additional Information: [full citation](#), [references](#), [citing](#), [index terms](#)

**Keywords:** O2, OpenGL, graphics hardware architecture, infinite-reality


### 3 [Tiled polygon traversal using half-plane edge functions](#)



Joel McCormack, Robert McNamara

August 2000 **Proceedings of the ACM SIGGRAPH/EUROGRAPHICS workshop on Graphics hardware**

Publisher: ACM Press

Full text available:  [pdf\(96.25 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citing](#), [index terms](#)

Existing techniques for traversing a polygon generate fragments one (or more) rows or columns at a time. (A fragment is all the information needed to paint one pixel of the polygon.) This order is non-optimal for many operations. For example, most frame buffers are tiled into rectangular pages, and there is a cost associated with accessing a different page. Pixel processing is more efficient if all fragments of a polygon on one page are generated before any fragments on a different page. Si ...

**Keywords:** graphics accelerators, rasterization, tiling

### 4 [A Characterization of Ten Hidden-Surface Algorithms](#)





Evan E. Sutherland, Robert F. Sproull, Robert A. Schumacker  
March 1974 **ACM Computing Surveys (CSUR)**, Volume 6 Issue 1

**Publisher:** ACM Press

Full text available: pdf(4.47 MB) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)



5 Parallel texture caching



Homan Igehy, Matthew Eldridge, Pat Hanrahan

July 1999 **Proceedings of the ACM SIGGRAPH/EUROGRAPHICS workshop on Graphics hardware**

**Publisher:** ACM Press

Full text available: pdf(1.80 MB) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)



6 Proceedings of the SIGNUM conference on the programming environment for development of numerical software



March 1979 **ACM SIGNUM Newsletter**, Volume 14 Issue 1

**Publisher:** ACM Press

Full text available: pdf(5.02 MB) Additional Information: [full citation](#)



7 Hybrid volume and polygon rendering with cube hardware



Kevin Kreeger, Arie Kaufman

July 1999 **Proceedings of the ACM SIGGRAPH/EUROGRAPHICS workshop on Graphics hardware**

**Publisher:** ACM Press

Full text available: pdf(1.85 MB) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)



**Keywords:** cube architecture, mixing polygons and volumes, ray casting, run-length-encoding, volume rendering

8 Hardware: Hexagonal storage scheme for interleaved frame buffers and textures



Yosuke Bando, Takahiro Saito, Masahiro Fujita

July 2005 **Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware HWWS '05**

**Publisher:** ACM Press

Full text available: pdf(703.31 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

This paper presents a storage scheme which statically assigns pixel/textel coordinates to multiple memory banks in order to minimize frame buffer and texture memory access load imbalance. In this scheme, the pixels stored in a particular memory bank are placed at the center and the vertices of hexagons packed in the frame buffer. By making these hexagons close to regular so that the pixel placement is uniform and isotropic, frame buffer and texture memory accesses are evenly distributed over the m ...



9 Identifying and Exploiting Spatial Regularity in Data Memory References

Tushar Mohan, Bronis R. de Supinski, Sally A. McKee, Frank Mueller, Andy Yoo, Martin Schulz  
November 2003 **Proceedings of the 2003 ACM/IEEE conference on Supercomputing**

**Publisher:** IEEE Computer Society

Full text available: pdf(264.75 KB) Additional Information: [full citation](#), [abstract](#)

The growing processor/memory performance gap causes the performance of many codes to be limited by memory accesses. If known to exist in an application, strided memory accesses forming streams can be targeted by optimizations such as prefetching, relocation, remapping, and vector loads. Undetected, they can be a significant source of memory stalls in loops. Existing stream-detection mechanisms either require special hardware, which may not gather statistics for subsequent analysis, or are limited ...



10 Dynamically scheduled VLIW processors

B. Ramakrishna Rau

December 1993 **Proceedings of the 26th annual international symposium on Microarchitecture**

Publisher: IEEE Computer Society Press

Full text available:  pdf(1.64 MB) Additional Information: [full citation](#), [references](#), [citations](#)

**Keywords:** VLIW processors, dynamic scheduling, multiple operation issue, out-of-order execution, scoreboarding


11 Execution-based prediction using speculative slices



Craig Zilles, Gurindar Sohi

May 2001 **ACM SIGARCH Computer Architecture News , Proceedings of the 28th annual international symposium on Computer architecture ISCA '01**, Volume 29 Issue 2

Publisher: ACM Press

Full text available:  pdf(1.03 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

*A relatively small set of static instructions has significant leverage on program execution performance. These problem instructions contribute a disproportionate number of cache misses and branch mispredictions because their behavior cannot be accurately anticipated using existing prefetching or branch prediction mechanisms.*

*The behavior of many problem instructions can be predicted by executing a small code fragment called a speculative slice. If a speculative slice is exec ...*

12 Zero-cycle loads: microarchitecture support for reducing load latency

Todd M. Austin, Gurindar S. Sohi

December 1995 **Proceedings of the 28th annual international symposium on Microarchitecture**

Publisher: IEEE Computer Society Press

Full text available:  pdf(1.35 MB) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

13 Artificial intelligence methods in process plant layout



A. McBrien, J. Madden, N. R. Shadbolt

June 1989 **Proceedings of the 2nd international conference on Industrial and engineering applications of artificial intelligence and expert systems - Volume 1 IEA/AIE '89**

Publisher: ACM Press

Full text available:  pdf(812.98 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

A constraint-based system, PLS, capable of deriving a technically acceptable three-dimensional layout for a chemical process plant of typical complexity is described. In the application of PLS to an exemplar plant, over five hundred instances of constraints were identified and manipulated. PLS has been implemented in a frame system created in-house. This features recursively structured frames to represent both the initial plant data and the intermediates derived by the system. Rea ...

14 Evaluating the impact of memory system performance on software prefetching and locality optimizations



Abdel-Hameed A. Badawy, Aneesh Aggarwal, Donald Yeung, Chau-Wen Tseng

June 2001 **Proceedings of the 15th international conference on Supercomputing**

Publisher: ACM Press

Full text available:  pdf(387.06 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)


Software prefetching and locality optimizations are techniques for overcoming the speed gap between processor and memory. In this paper, we evaluate the impact of memory

trends on the effectiveness of software prefetching and locality optimizations for three types of applications: regular scientific codes, irregular scientific codes, and pointer-chasing codes. We find for many applications, software prefetching outperforms locality optimizations when there is sufficient memory bandwidth, but ...

15 Pomegranate: a fully scalable graphics architecture

 Matthew Eldridge, Homan Igehy, Pat Hanrahan  
July 2000 **Proceedings of the 27th annual conference on Computer graphics and interactive techniques**


**Publisher:** ACM Press/Addison-Wesley Publishing Co.

Full text available:  pdf(508.39 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Pomegranate is a parallel hardware architecture for polygon rendering that provides scalable input bandwidth, triangle rate, pixel rate, texture memory and display bandwidth while maintaining an immediate-mode interface. The basic unit of scalability is a single graphics pipeline, and up to 64 such units may be combined. Pomegranate's scalability is achieved with a novel "sort-everywhere" architecture that distributes work in a balanced fashion at every stage of the pipeline, ke ...

**Keywords:** graphics hardware, parallel computing

16 Compiler-directed high-level energy estimation and optimization

 I. Kadayif, M. Kandemir, G. Chen, N. Vijaykrishnan, M. J. Irwin, A. Sivasubramaniam  
November 2005 **ACM Transactions on Embedded Computing Systems (TECS)**, Volume 4 Issue 4

**Publisher:** ACM Press

Full text available:  pdf(891.31 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)


The demand for high-performance architectures and powerful battery-operated mobile devices has accentuated the need for power optimization. While many power-oriented hardware optimization techniques have been proposed and incorporated in current systems, the increasingly critical power constraints have made it essential to look for software-level optimizations as well. The compiler can play a pivotal role in addressing the power constraints of a system as it wields a significant influence on the ...

**Keywords:** Energy-Aware Compilation (EAC), mobile devices

17 Parsing: An HPSG parser based on Description Logics


J. Joachim Quantz  
August 1994 **Proceedings of the 15th conference on Computational linguistics - Volume 1**

**Publisher:** Association for Computational Linguistics

Full text available:  pdf(452.17 KB) Additional Information: [full citation](#), [abstract](#), [references](#)

In this paper I present a parser based on Description Logics (DL) for a German HPSG-Style fragment. The specified parser relies mainly on the inferential capabilities of the underlying DL system. Given a preferential default extension for DL disambiguation is achieved by choosing the parse containing a qualitatively minimal number of exceptions.

18 Characterizing the caching and synchronization performance of a multiprocessor operating system

 Josep Torrellas, Anoop Gupta, John Hennessy  
September 1992 **ACM SIGPLAN Notices , Proceedings of the fifth international conference on Architectural support for programming languages and operating systems ASPLOS-V**, Volume 27 Issue 9

**Publisher:** ACM Press

Full text available:  pdf(1.52 MB) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

19 A fast and accurate framework to analyze and optimize cache memory behavior



Xavier Vera, Nerina Bermudo, Josep Llosa, Antonio González

March 2004 **ACM Transactions on Programming Languages and Systems (TOPLAS)**,

Volume 26 Issue 2

**Publisher:** ACM Press

Full text available: pdf(270.06 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#),  
[review](#)

The gap between processor and main memory performance increases every year. In order to overcome this problem, cache memories are widely used. However, they are only effective when programs exhibit sufficient data locality. Compile-time program transformations can significantly improve the performance of the cache. To apply most of these transformations, the compiler requires a precise knowledge of the locality of the different sections of the code, both before and after being transformed. Cache ...

**Keywords:** Cache memories, optimization, sampling

## 20 [Reducing instruction cache energy consumption using a compiler-based strategy](#)



W. Zhang, J. S. Hu, V. Degalahal, M. Kandemir, N. Vijaykrishnan, M. J. Irwin

March 2004 **ACM Transactions on Architecture and Code Optimization (TACO)**, Volume 1  
Issue 1

**Publisher:** ACM Press

Full text available: pdf(1.15 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Excessive power consumption is widely considered as a major impediment to designing future microprocessors. With the continued scaling down of threshold voltages, the power consumed due to leaky memory cells in on-chip caches will constitute a significant portion of the processor's power budget. This work focuses on reducing the leakage energy consumed in the instruction cache using a compiler-directed approach. We present and analyze two compiler-based strategies termed as conservative and optim ...

**Keywords:** Leakage power, cache design, compiler optimizations

Results 1 - 20 of 30

Result page: [1](#) [2](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2006 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads: [Adobe Acrobat](#) [QuickTime](#) [Windows Media Player](#) [Real Player](#)

☐ Search Results

[BROWSE](#)

[SEARCH](#)

[IEEE XPLORE GUIDE](#)

[SUPPORT](#)

Results for "(((shading <and> fragment <and> conflict <and> tile))<in>metadata)"

Your search matched 0 documents.

A maximum of 100 results are displayed, 25 to a page, sorted by **Relevance** in **Descending** order.

☒ e-mail  printer friendly

» Search Options

[View Session History](#)

[New Search](#)

Modify Search

(((shading <and> fragment <and> conflict <and> tile))<in>metadata)

**Search** 

☐ Check to search only within this results set

Display Format: ☒ Citation ☐ Citation & Abstract

» Key

IEEE JNL IEEE Journal or Magazine

IEE JNL IEE Journal or Magazine

IEEE CNF IEEE Conference Proceeding

IEE CNF IEE Conference Proceeding

IEEE STD IEEE Standard

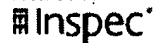
**No results were found.**

Please edit your search criteria and try again. Refer to the Help pages if you need assistance revising your search.

[Help](#) [Contact Us](#) [Privacy & Security](#) [IEEE.org](#)

© Copyright 2006 IEEE – All Rights Reserved

Indexed by



**Search Results**

[BROWSE](#)

[SEARCH](#)

[IEEE XPLORE GUIDE](#)

[SUPPORT](#)

Results for "(((shader <and> fragment <and> conflict <and> tile))<in>metadata)"

Your search matched **0** documents.

A maximum of 100 results are displayed, 25 to a page, sorted by **Relevance** in **Descending** order.

 [e-mail](#)  [printer friendly](#)

» **Search Options**

[View Session History](#)

[New Search](#)

**Modify Search**

(((shader <and> fragment <and> conflict <and> tile))<in>metadata)

**Search** 

☐ Check to search only within this results set

Display Format: ☒ Citation ☐ Citation & Abstract

» **Key**

IEEE JNL IEEE Journal or Magazine

IEE JNL IEE Journal or Magazine

IEEE CNF IEEE Conference Proceeding

IEE CNF IEE Conference Proceeding

IEEE STD IEEE Standard

**No results were found.**

Please edit your search criteria and try again. Refer to the Help pages if you need assistance revising your search.

[Help](#) [Contact Us](#) [Privacy & Security](#) [IEEE.org](#)

© Copyright 2006 IEEE – All Rights Reserved

Indexed by



## RESULT LIST

0 results found in the Worldwide database for:

**conflict AND shader AND fragment AND tile** in the title or abstract

(Results are sorted by date of upload in database)

---

Data supplied from the *esp@cenet* database - Worldwide

## RESULT LIST

0 results found in the Worldwide database for:

**conflict AND shading AND fragment AND tile** in the title or abstract

(Results are sorted by date of upload in database)

---

Data supplied from the *esp@cenet* database - Worldwide



## RESULT LIST

0 results found in the Worldwide database for:

**stall AND shading AND fragment AND tile** in the title or abstract

(Results are sorted by date of upload in database)

---

Data supplied from the *esp@cenet* database - Worldwide

## RESULT LIST

0 results found in the Worldwide database for:

**delay AND shading AND fragment AND tile** in the title or abstract

(Results are sorted by date of upload in database)

---

Data supplied from the *esp@cenet* database - Worldwide

# Searching PAJ

[MENU](#)[NEWS](#)[HELP](#)

Search Results : 0

[Clear](#)**Text Search**

If you want to conduct a Number Search, please click on the button to the right.

[Number Search](#)

**Applicant, Title of invention, Abstract** --- e.g. computer semiconductor

If you use the AND/OR operation, please leave a SPACE between keywords.

One letter word or Stopwords are not searchable.

fragment tile conflict

AND

AND

shading shader

OR

AND

delay stall postpone

OR

AND

**Date of publication of application** --- e.g. 19980401 - 19980405

AND

**IPC** --- e.g. D01B7/04 A01C11/02

If you use the OR operation, please leave a SPACE between keywords.

Search

Stored data

Copyright (C); 1998,2003 Japan Patent Office

# Searching PAJ

[MENU](#)[NEWS](#)[HELP](#)

Search Results : 0

[Clear](#)**Text Search**

If you want to conduct a Number Search, please click on the button to the right.

[Number Search](#)

**Applicant, Title of invention, Abstract** --- e.g. computer semiconductor

If you use the AND/OR operation, please leave a SPACE between keywords.

One letter word or Stopwords are not searchable.

fragment tile

AND ▼

AND

shading shader

OR ▼

AND

delay stall postpone

OR ▼

AND

**Date of publication of application** --- e.g. 19980401 - 19980405

-

AND

**IPC** --- e.g. D01B7/04 A01C11/02

If you use the OR operation, please leave a SPACE between keywords.



Search

Stored data

Copyright (C); 1998,2003 Japan Patent Office